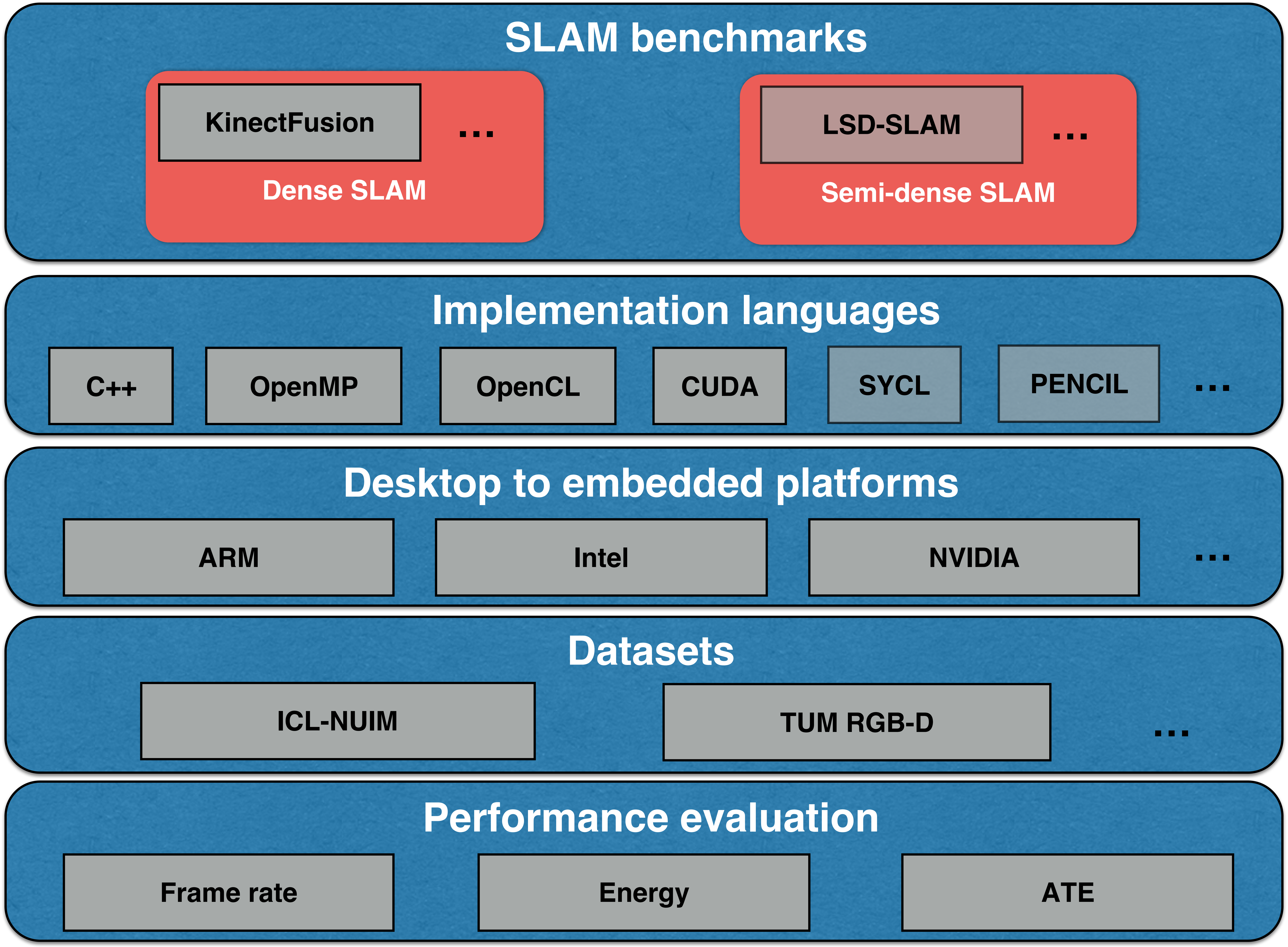
Comparative design space exploration of dense and semi-dense SLAM

Z. Zia, L. Nardi, A. Jack, E. Vespa, B. Bodin, P. H. J. Kelly, A. J. Davison

Imperial College London

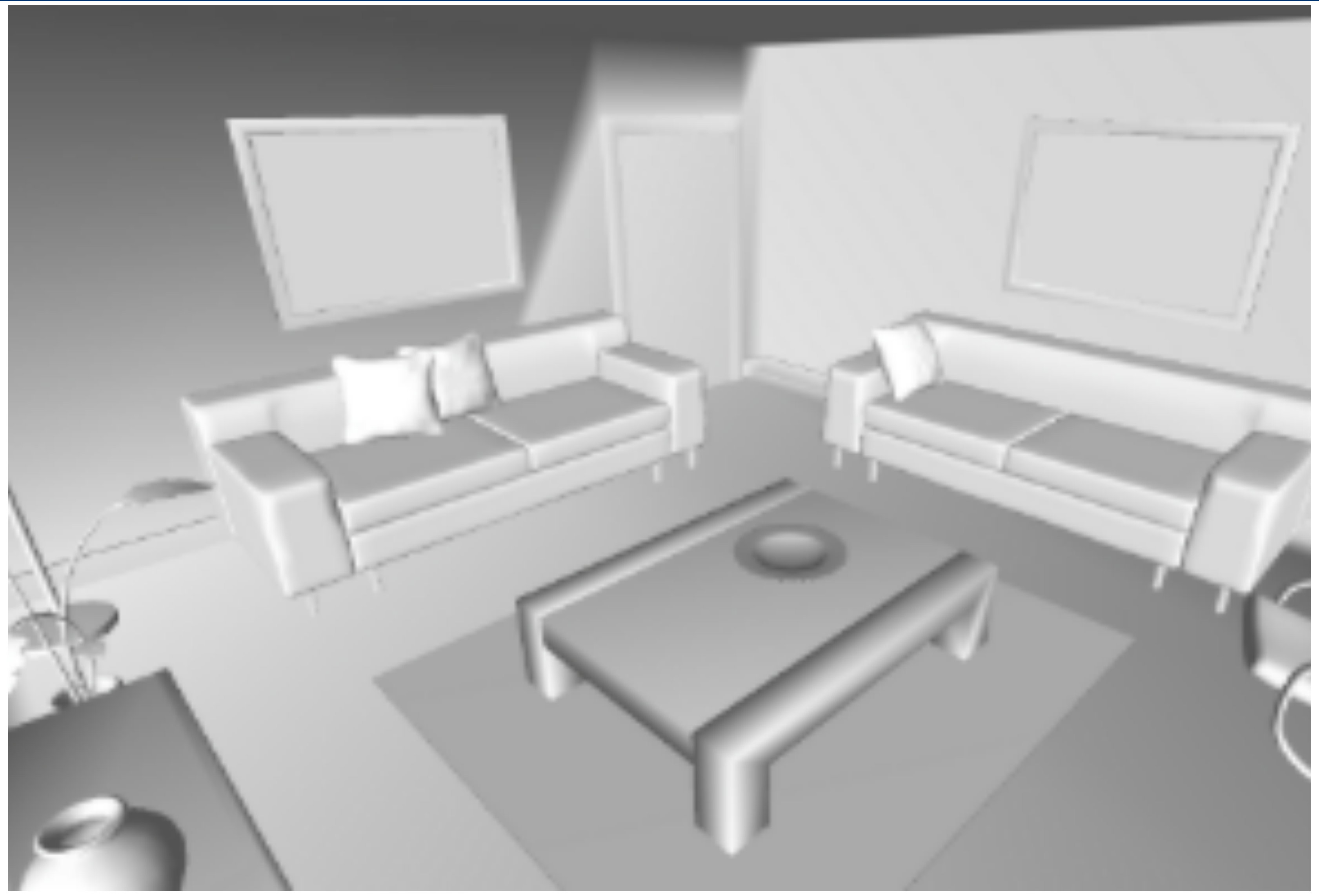
The SLAMBench benchmarking framework

- Enabling end-to-end quantitative and reproducible benchmarking of SLAM pipelines
- SLAM as a multi-objective optimisation problem
  - Absolute Trajectory Error (ATE)
  - Relative Pose Error (RPE)
  - Frame rate
  - Energy per frame
  - Reconstruction accuracy (coming…)



Datasets

- ICL-NUIM synthetic indoor scenes:
  - Living room synthetic environment
  - Human generated trajectories
  - Trajectory *and* world model ground-truth
  - A. Handa *et al*. A Benchmark for RGB-D Visual Odometry, 3D Reconstruction and SLAM.



ICL-NUIM living room http://www.doc.ic.ac.uk/~ahanda/VaFRIC/iclnuim.html

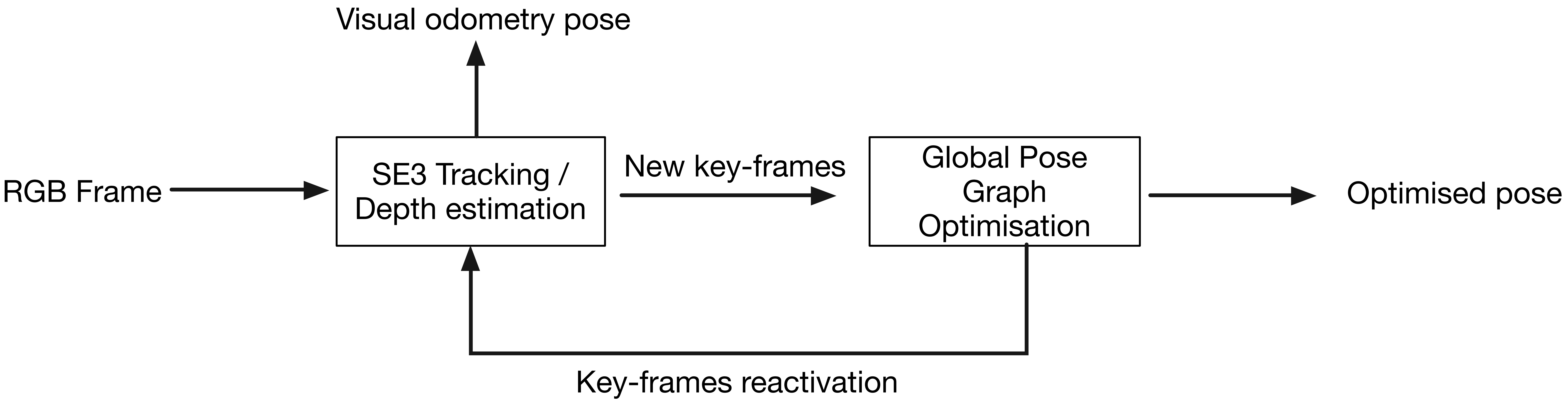
- TUM real RGB-D dataset:
  - Handheld camera sequence plus trajectory ground-truth
  - J. Sturm *et al*. A benchmark for the evaluation of RGB-D SLAM systems

Screenshot from TUM fr2/desk sequence

LSD-SLAM pipeline structure

Imperial College London

- Semi-dense tracking and mapping front-end
  - Track new images against high gradient patches of reference key-frame
  - Estimate and refine depth for such patches
  - When current frame too far, finalise key-frame and initialise a new one
- Pose graph optimisation on back-end
  - Loop closure detection
  - Global optimisation



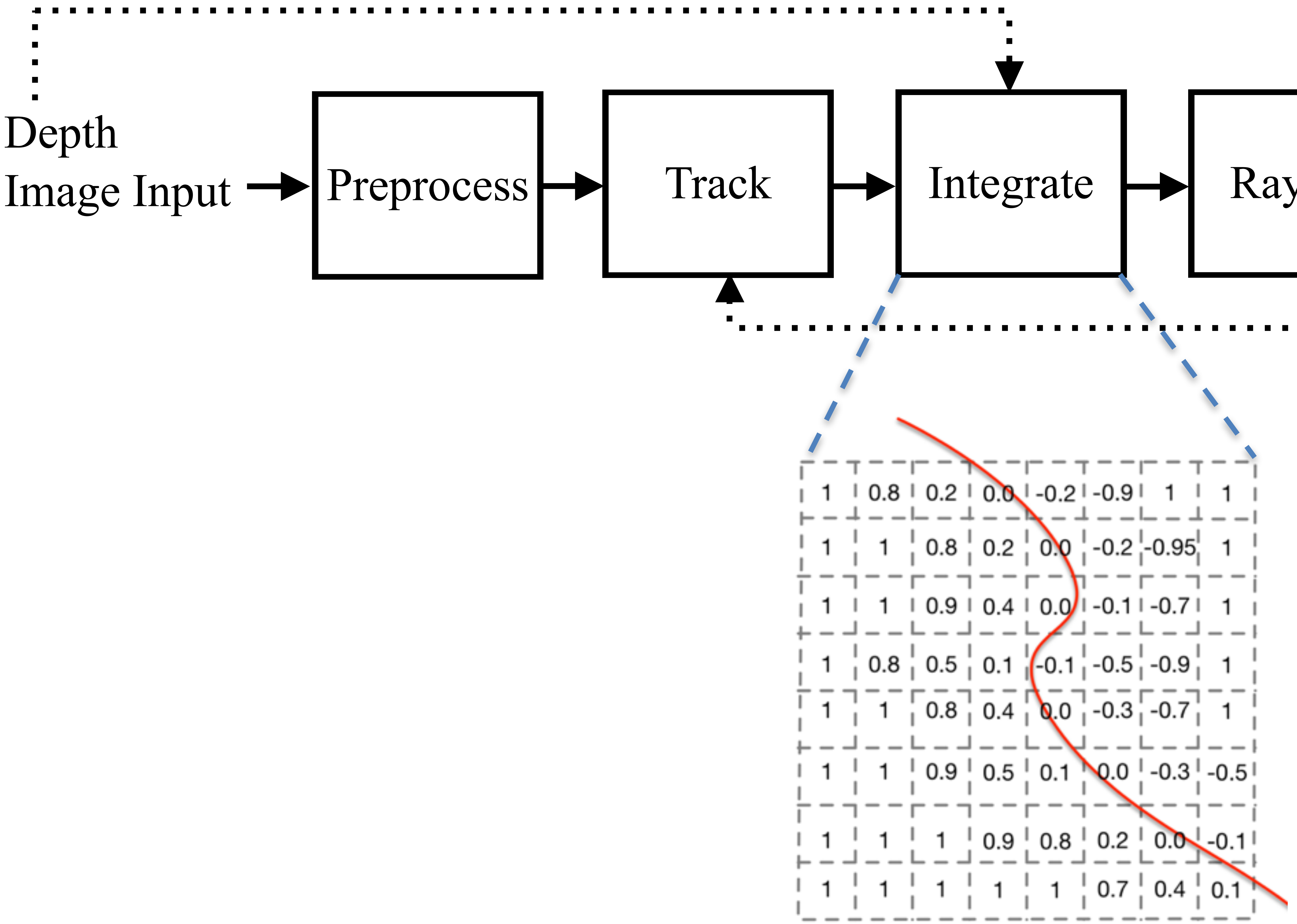
LSD-SLAM kernels

Imperial College London

Thread name	Major kernels	Description	Pattern	Percent
Tracking (vectorized)	*Calc. Residuals*	Calculate components of the Levenberg–Marquardt (LM) algorithm	Map	72%
	Calc. Weights and Residuals		Map	4%
	Calc. Jacobians		Map-Reduce	9%
	Solve	Evaluate the LM algorithm given the above calculations	External	0%
Total				**34 s**
Depth	*Stereo Line Search*	Epipolar line search	Map	43%
	Fill Holes	Increase density of depth map	Stencil	20%
	Regularize Depth Map	Denoise the depth map	Stencil	28%
	Copy Depth Map to Frame	Implementation specific overhead	Map	6%
Total				**48 s**
Constraint Search	*Find Euclidean Overlaps*	Get neighbour frames from graph, to insert new constraints	Search	6%
	Filter and Sorting	Remove less optimal frames from results	Map	4%
	Calc. Residuals	Calculate components of the Levenberg—Marquardt (LM) algorithm between keyframe and neighbour frames	Map	71%
	Calc. Weights and Residuals		Reduce	7%
	Calc. Jacobian Matrix		External	12%
Total				**19 s**
Optimization	*g2o Call*	Run iterations of global optimization	External	99%
	Update Graph	Incorporate improvements from g2o into graph	Map	1%
Total				**3 s**

KinectFusion pipeline

- Dense geometry estimation encoded in a *truncated signed-distance function* (TSDF)
- Dense tracking via frame-to-model alignment: synthetic point cloud obtained by ray-casting the TSDF



Truncated signed-distance function

The red line shows the zero iso-surface representing the best estimate of the observed surface

KinectFusion kernels

Major kernels	Block	Pattern	Percent
Convert mm to meters	Preprocess	Gather	0%
Bilateral Filter	Preprocess	Stencil	4%
Half Sample	Track	Stencil	0%
Depth to Vertex		Map	0%
Vertex to Normal		Stencil	0%
Track		Map/Gather	2%
Reduce		Reduction	2%
Solve		Sequential	0%
Integrate	Integrate	Map/Gather	73%
Raycast	Raycast	Search/Stencil	17%

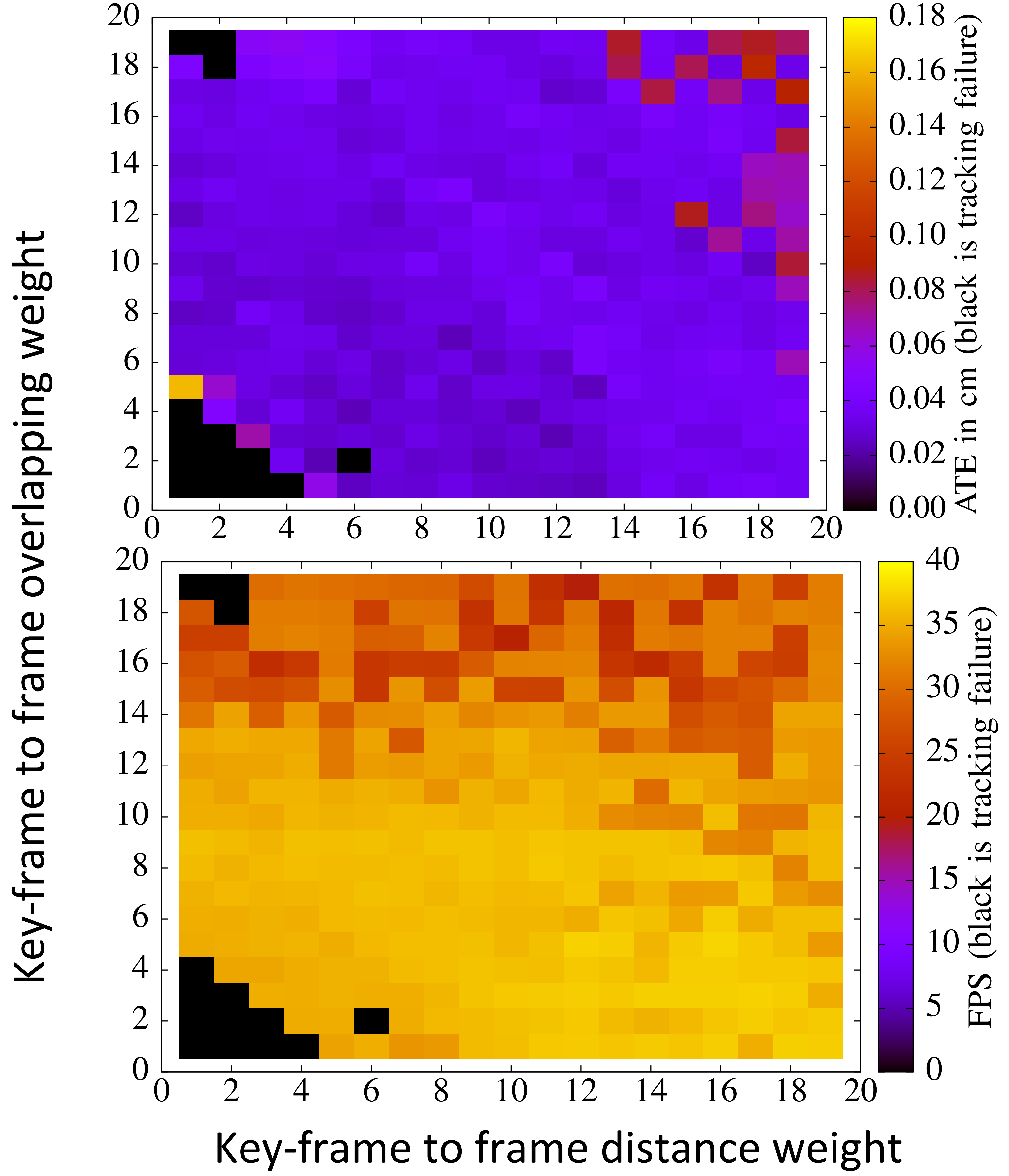
LSD-SLAM and KinectFusion design-space exploration

- LSD-SLAM:
  - Number of key-frames on ATE and frame-rate
  - Depth map density
  - Hardware characteristics (frequency + number/type of cores)
- KinectFusion:
  - ATE versus voxel size
  - Frame-rate versus voxel size
  - Hardware characteristics (frequency + number/type of cores)

LSD-SLAM design space exploration with SLAMBench

Imperial College London

- Impact of key-frames number on ATE and frame-rate.
- Parametric weights that dictate how often new key-frames are created:
  - X axis: weight assigned to Euclidean distance between current frame and reference key-frame
  - Y axis: weight assigned to current frame and reference key-frame overlapping
- Higher values imply more key-frames.
- Black regions represent configurations that make the algorithm lose track

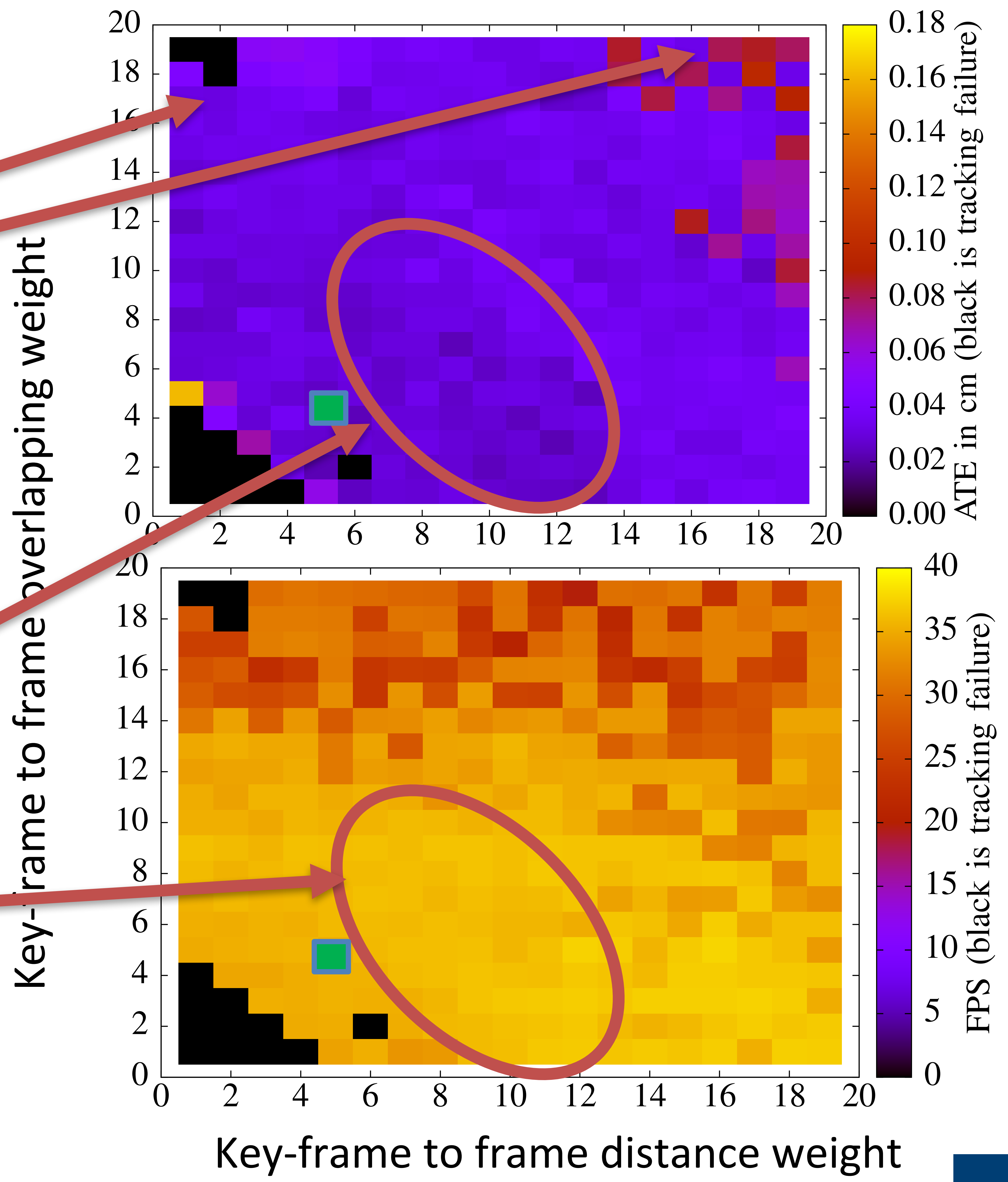


LSD-SLAM design space exploration with SLAMBench

- Impact of key-frames number on ATE and frame-rate
- Parametric weights:
  - Euclidean distance
  - Frame to key-frame overlapping
- Default configuration

Too many, poor depth refinements imply bad ATE

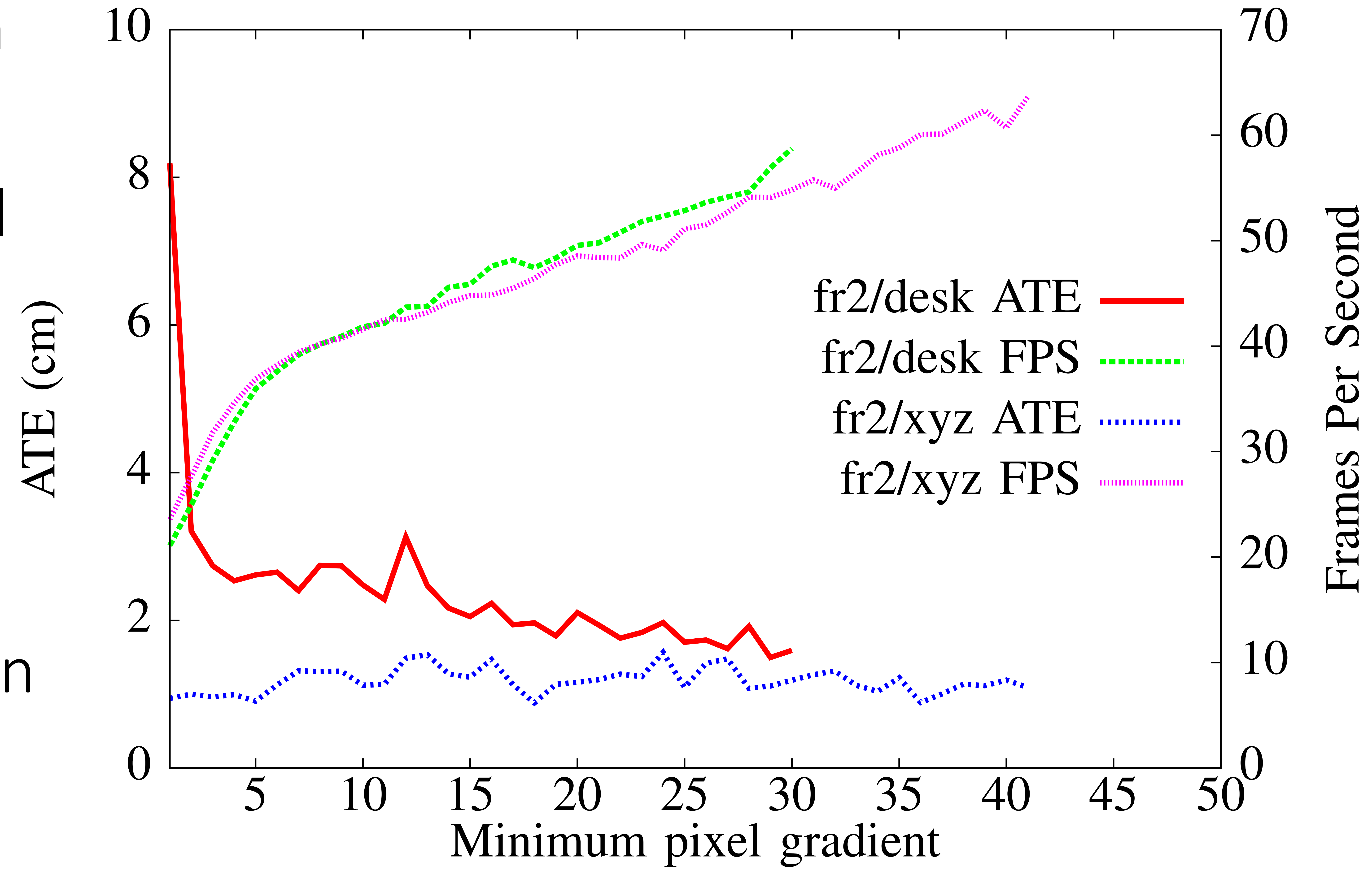
Regions were you attain best ATE and frame-rate



LSD-SLAM design-space exploration with SLAMBench

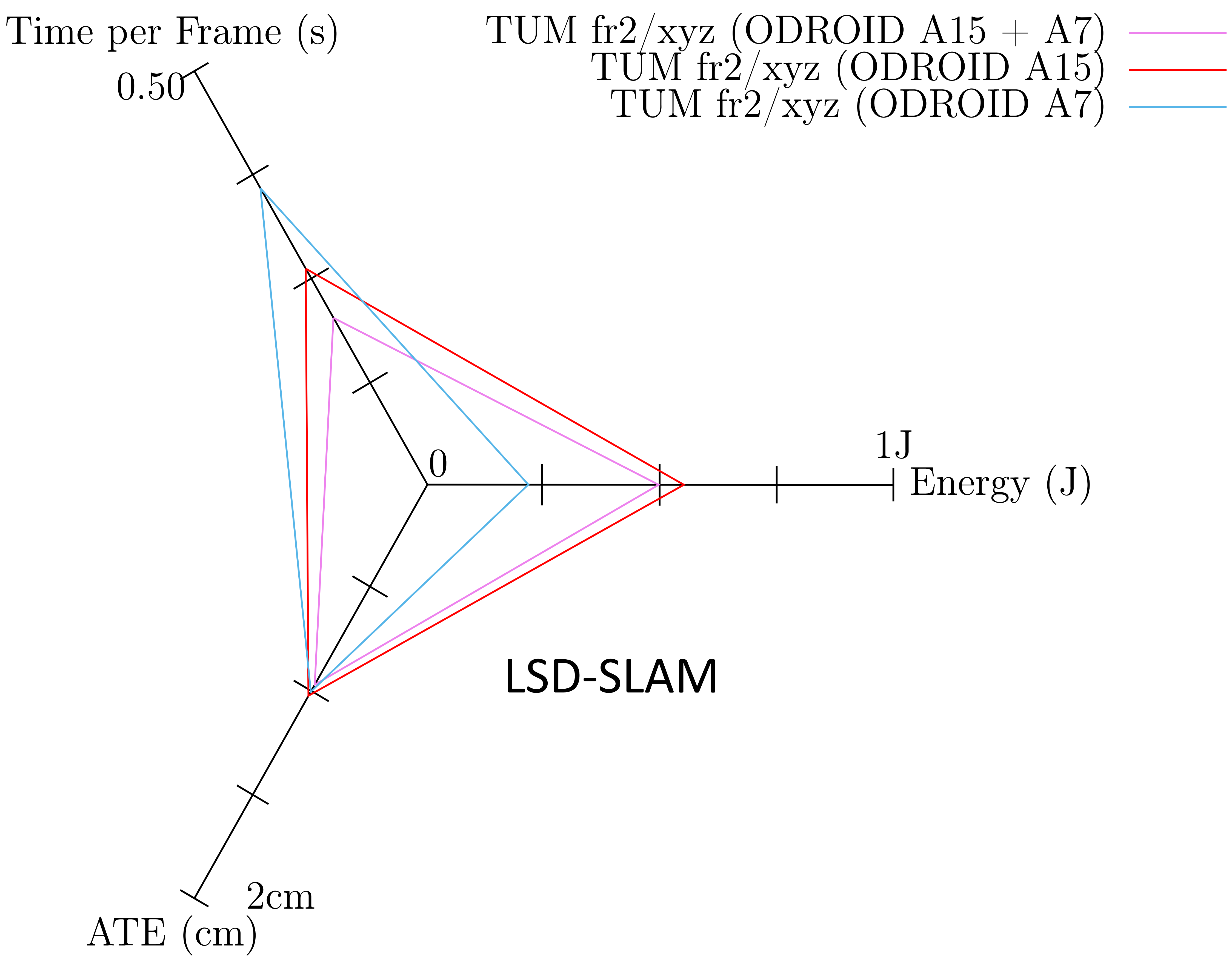
Imperial College London

- LSD-SLAM depth estimation
  - Higher gradient threshold implies less pixels selected for tracking and mapping
  - Higher frame-rate given from the reduced number of epipolar searches
- Accuracy heavily depends on sequence.

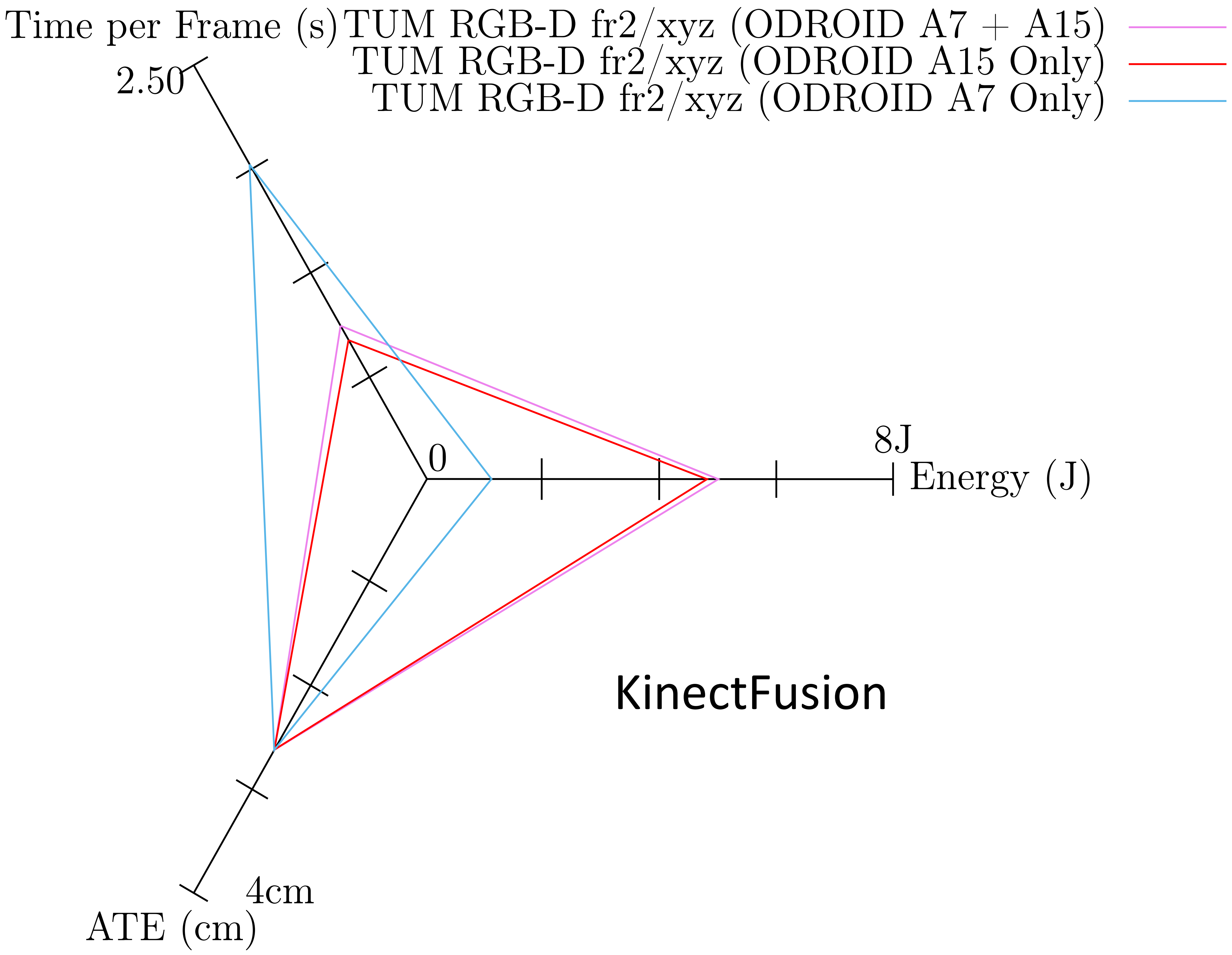


LSD-SLAM: hardware configuration exploration

- Hardware configurations exploration on the ODROID board
  - ARM big.LITTLE architecture: 4 A7 + 4 A15 cores
  - Holistic comparison varying the number of cores



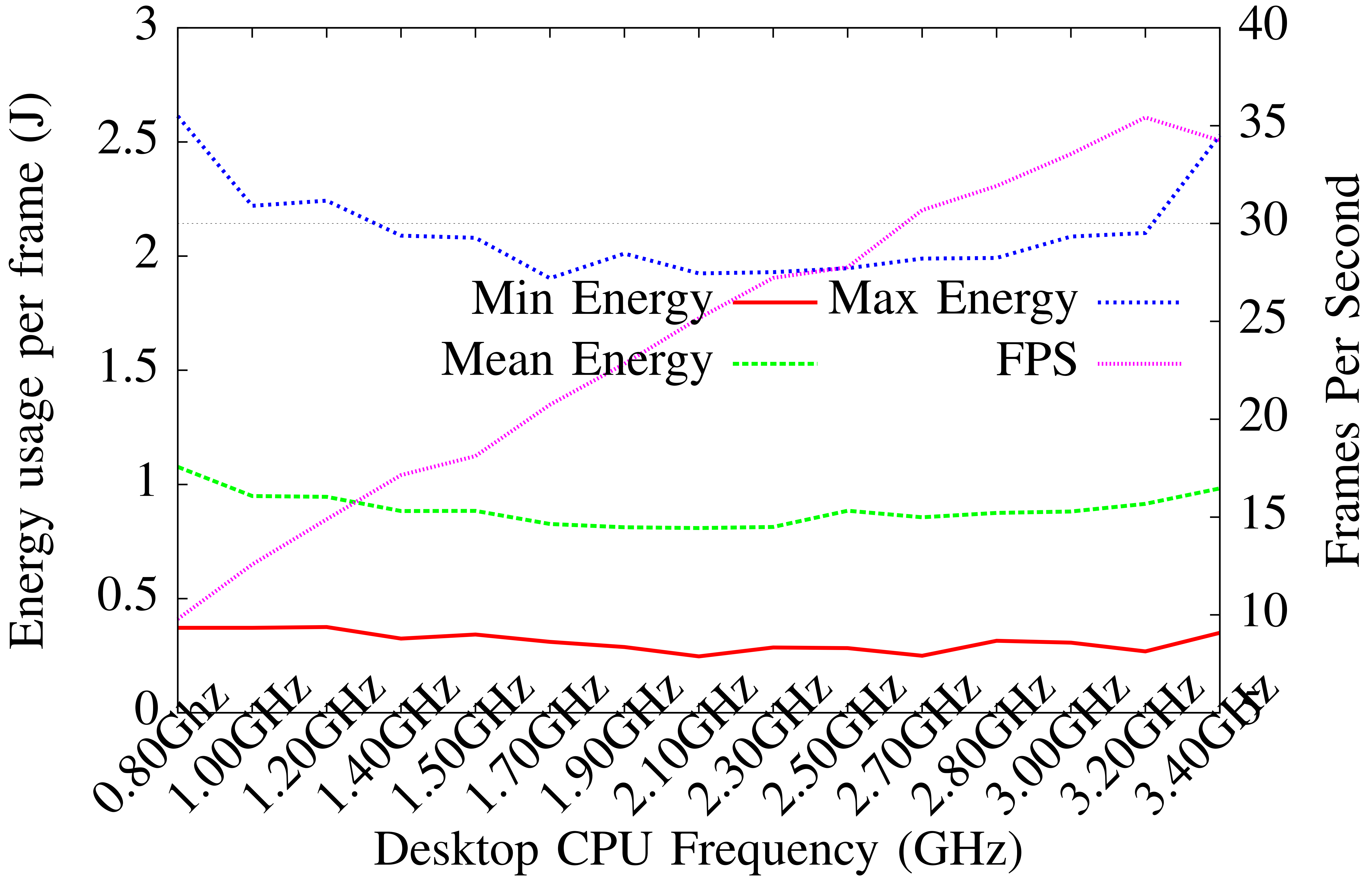
LSD-SLAM



KinectFusion

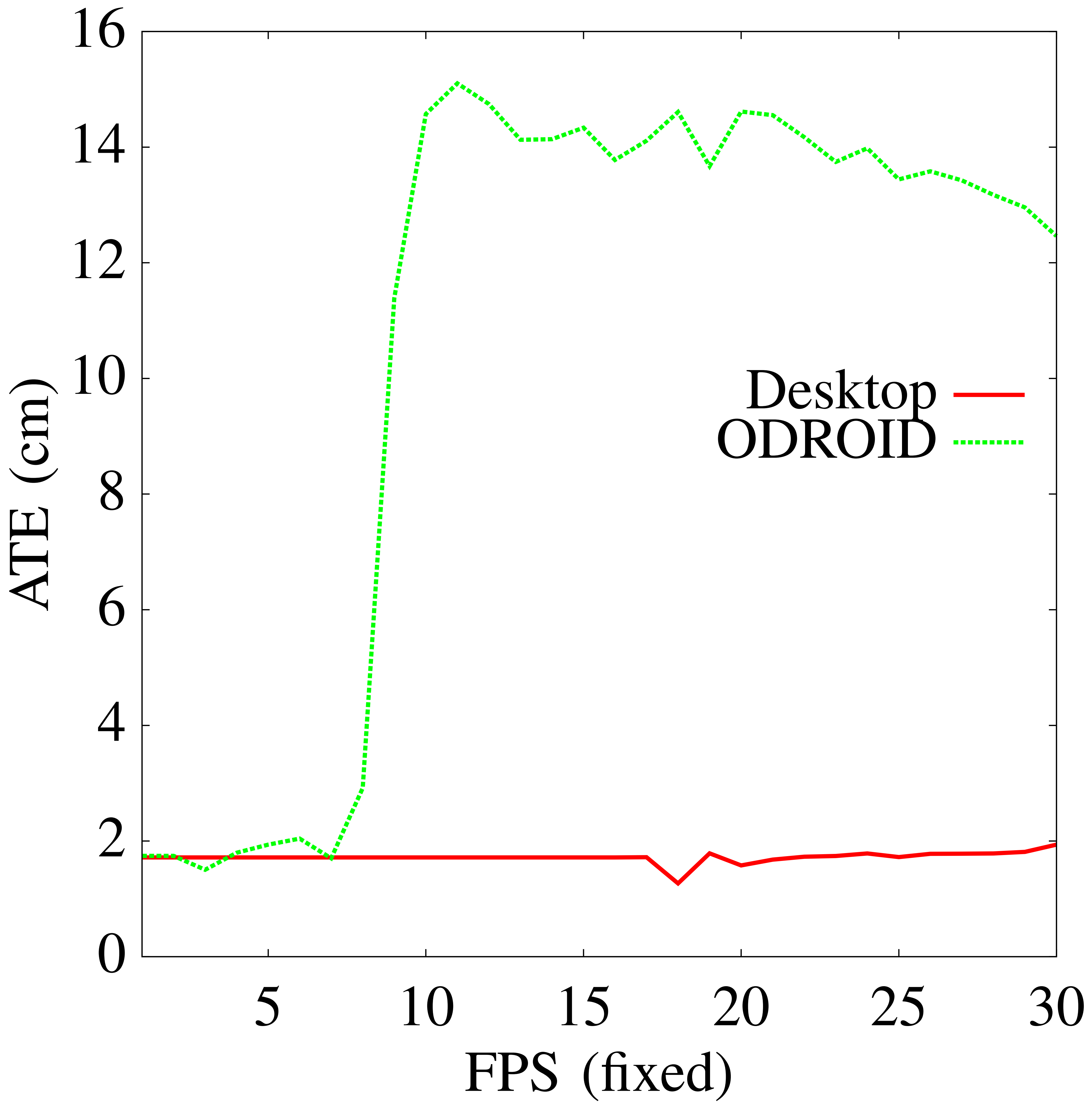
LSD-SLAM: hardware configuration exploration

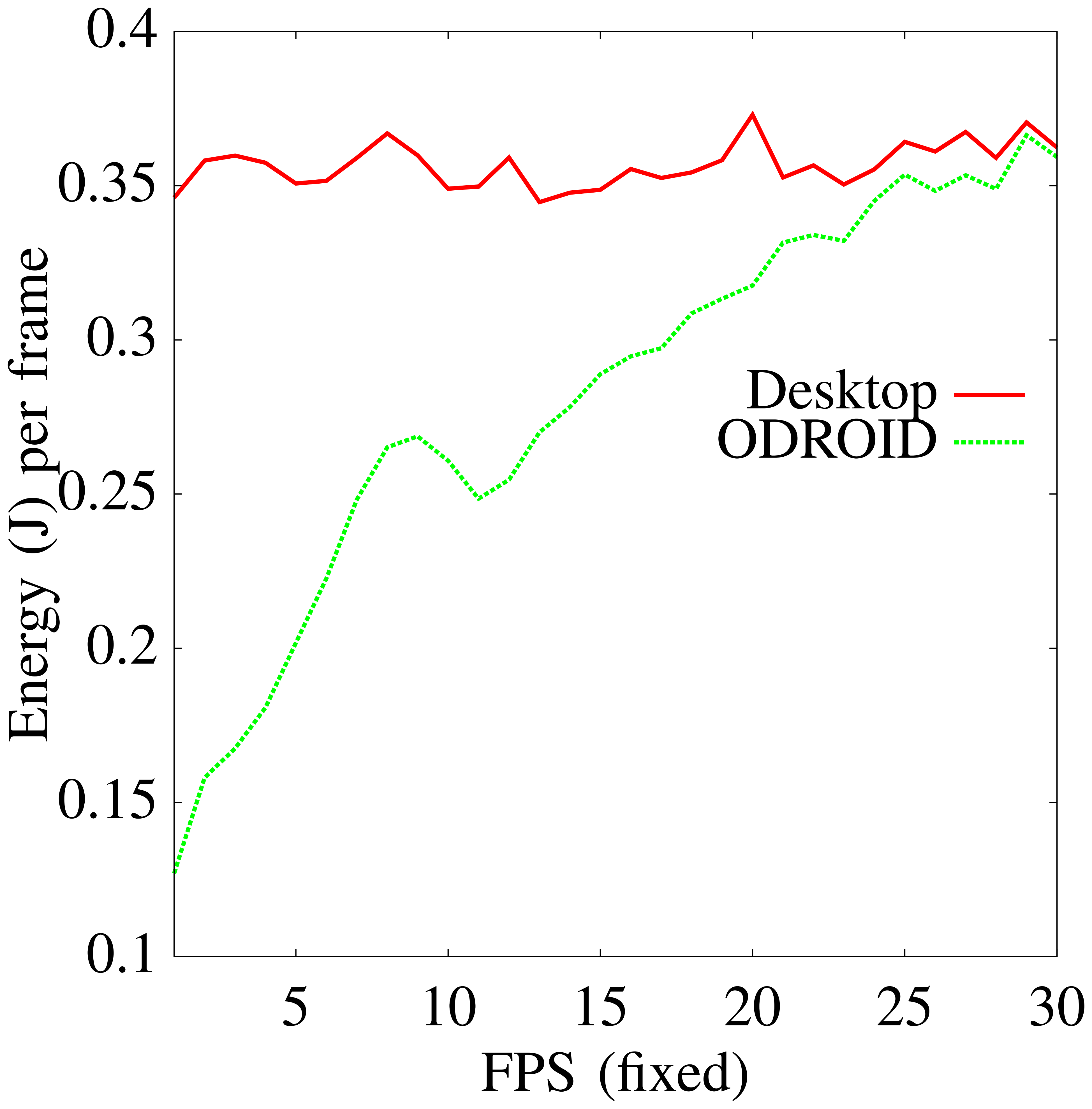
- Frequency scaling on a Haswell i7-4770 desktop processor
  - Mean energy per frame stays constant, frame rate increases sub-linearly



LSD-SLAM: frame-rate impact on tracking accuracy

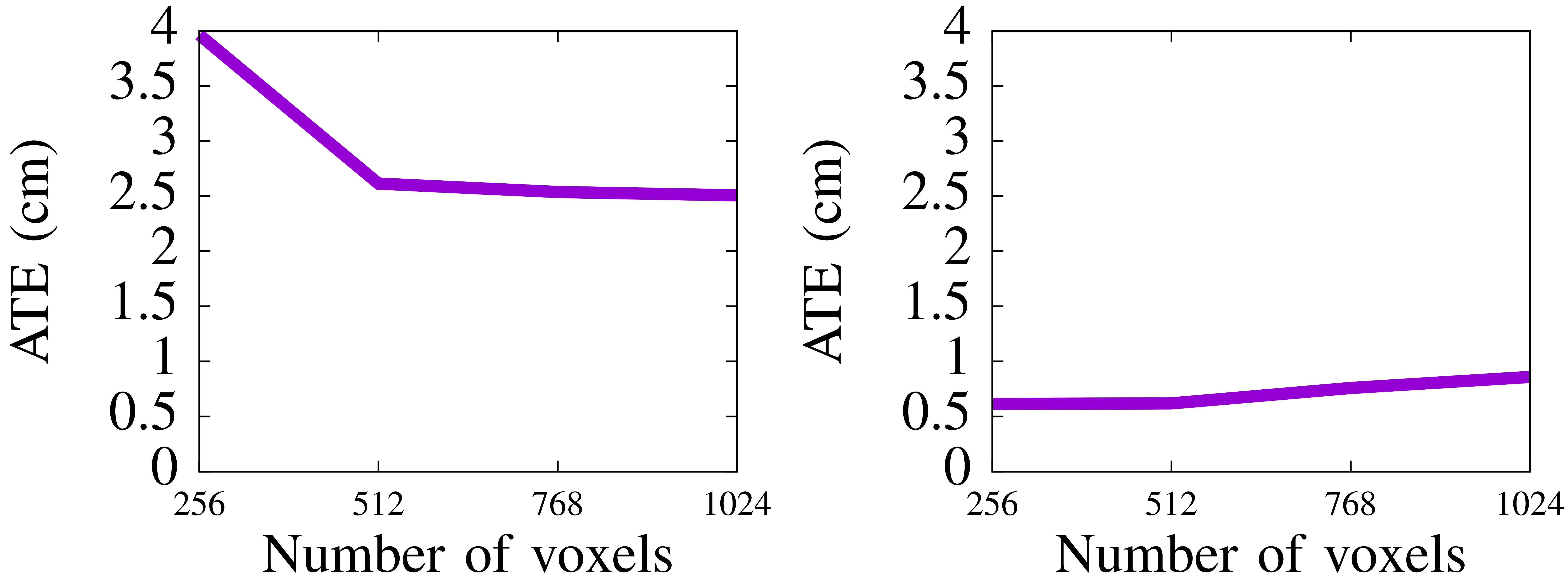
- Removing the constraing of process – every-frame mode
  - Fix frame rate - frames can be dropped
- Test platforms:
  - ODROID board (A7 + A15 ARM cores)
  - Desktop: Intel Haswell i7 4770
- Interesting impact of frame-rate on LSD-SLAM accuracy on the ODROID board
  - Frame dropping considerably impacts tracking accuracy





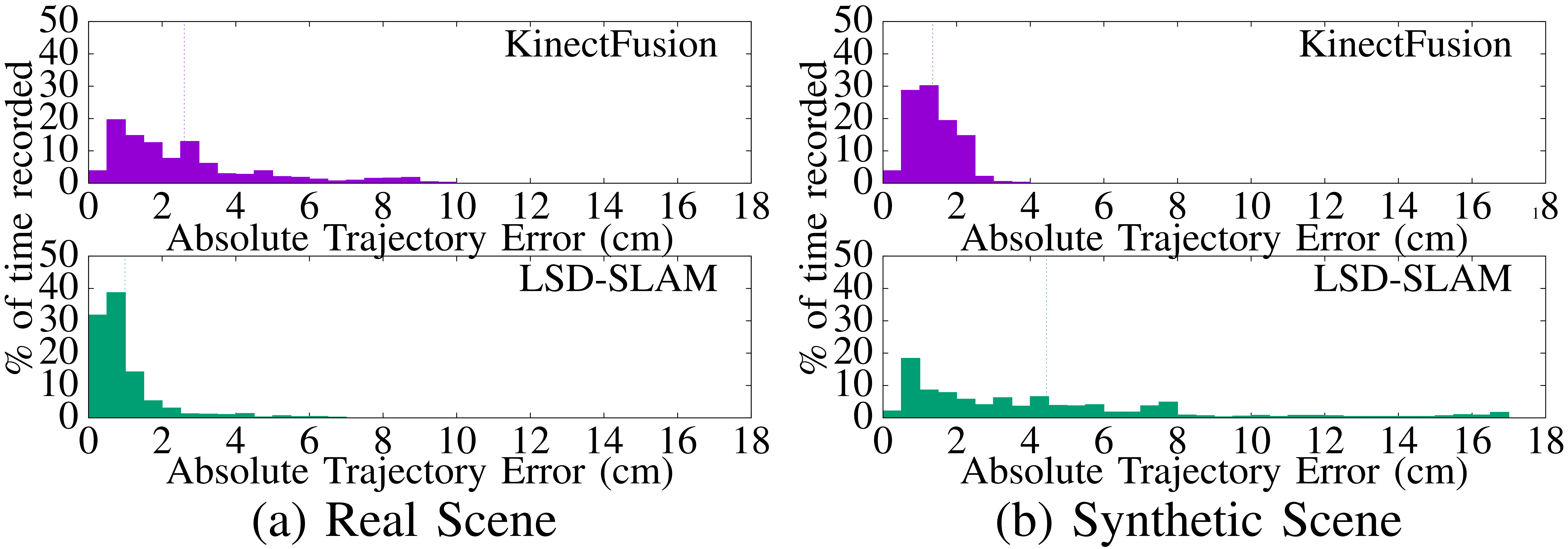
KinectFusion: accuracy versus resolution

- Scaling up the resolution does not always imply a better accuracy
  - Coarser voxels might have a noise smoothing effect leading to better tracking



(a) Real Scene (b) Synthetic Scene

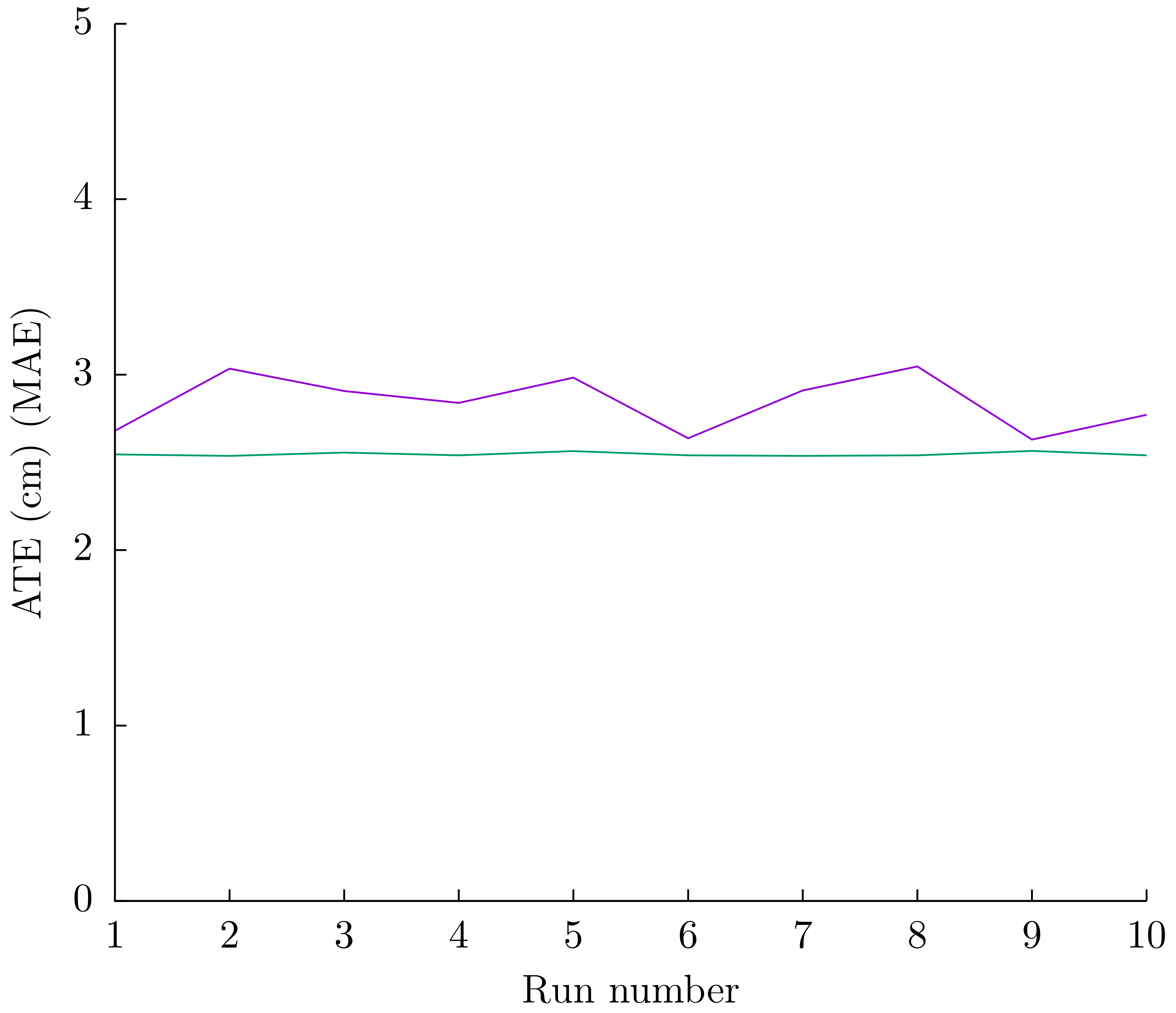
Comparing LSD-SLAM and KinectFusion



- Comparing KinectFusion and LSD-SLAM
  - Absolute trajectory error distribution over entire trajectory
  - Real scene vs synthetic scene
  - LSD-SLAM possibly affected by lack of realism in synthetic RGB data

A note on result reproducibility

- In this work we enforced *process-every-frame* mode for reproducibility purposes
- LSD-SLAM exposed significant fluctuations across repeated executions



Acknowledgements

Imperial College London

- We would like to thank EPSRC for funding this research, PAMELA grant EP/K008730.
- Jacob Engel for useful discussion and feedback on LSD-SLAM
- Andy Nisbet and John Mawyer of University of Manchester for their contributions to the SLAMBench framework